IN THE SPECIFICATION

Please amend the specification as detailed below:

Please change the first paragraph of page 9 to read as shown below:

These and other objects, features and advantages of the present invention will be more readily apparent from the following description of one ~~the~~ preferred embodiment of the invention in which:

Please change the description of FIG. 5A, FIG. 5B and 5C on page 9 to read as shown below:

FIG. 5A, FIG. 5B and FIG. 5C ~~5B~~ are block diagrams showing the sequence of steps in the present invention used for extracting, aggregating and storing information utilized in system processing from: user input, the process management system database, optionally, the simulation program database; the Internet; and the Owner Value and RiskMap® System database;

Please change the description of FIG. 6A and 6B on page 9 to read as shown below:

FIG. 6A, ~~and~~ FIG. 6B, FIG. 6C, FIG. 6D, FIG. 6E and FIG. 6F are block diagrams showing the sequence of steps in the present invention that are utilized in identifying the process features that maximizes expected process value while minimizing risk for the enterprise or multi-enterprise organization that owns the process;

Please change the caption and first paragraph of page 10 to read as shown below:

DETAILED DESCRIPTION OF ONE ~~THE~~ PREFERRED EMBODIMENT

FIG.1 provides an overview of the processing completed by the innovative system for process management. In accordance with the present invention, an automated method of and system (100) for optimizing risk and return from a process is provided. Processing starts in this system

(100) with a block of software (200) that extracts, aggregates and stores the data and user input required for completing the analysis. This information is extracted via a network (25) from a process management system database (30), optionally, a simulation program database (35), the Internet (40) and an Owner Value and RiskMap® System database (45). There are also optional data extractions from a Owner Basic Financial System database (6), a Owner Advanced Financial System database (7), a Owner Operations System database (8) and one or more Owner Asset System database(s) (9). These information extractions and aggregations are guided by a user (20) through interaction with a user-interface portion of the application software (900) that mediates the display and transmission of all information to the user (20) from the system (100) as well as the receipt of information into the system (100) from the user (20) using a variety of data windows tailored to the specific information being requested or displayed in a manner that is well known. While only one database of each type (30, 35 & 45) is shown in FIG. 1, it is to be understood that the system (100) can extract data from multiple databases of each type via the network (25).

Please change the second paragraph of page 10 to read as shown below:

All extracted information concerning the process is stored in a file or table (hereinafter, table) within an application database (50) as shown in FIG. 2. The application database (50) contains tables for storing user input, extracted information and system calculations including a system settings table (140), a metadata mapping table (141), a conversion rules table (142), a frame definition table (143), a process management system database table (144), a reports table (145), a process to owner table (146), an operating factors table (147), a simulation program table (148), a bot date table (149), an Owner Value and RiskMap™ System table (150), a process value table (151), a external factor forecast table (152), a feature option value table (153), a sensitivity analysis table (154), an optimal risk profile table (155) and an analysis definition table (156). The application database also optionally has an advanced finance system table (157), analysis definition table (158), an asset system table (159), a basic financial system table (160), a derivative table (161), an element/external factor definition table (162), and element variables table (163), an enterprise sentiment table (164), an external database table (165), an xml summary table (166), a factor variables table (167), a financial forecast table (168), a generic risk table (169), an industry ranking table (170), an operation systems table (171), an optimal mix table (172), a real option value table (173), a risk reduction activity/product

table (174), a scenarios table (175), a segment definition table (176), a simulations table (177) a statistics table (178) and a vector table (179~)~ The application database (50) can optionally exist as a datamart, data warehouse, departmental warehouse or storage area network. The system of the present invention has the ability to accept and store supplemental or primary data directly from user input, a data warehouse or other electronic files in addition to receiving data from the databases described previously. The system of the present invention also has the ability to complete the necessary calculations without receiving data from one or more of the specified databases. However, in one the preferred embodiment all required information is obtained from the specified databases (30, 35 & 45) and the Internet (40).

Please change the paragraph that begins on line 7 of page 11 to read as shown below:

As shown in FIG. 3, one the preferred embodiment of the present invention is a computer system (100) illustratively comprised of a client personal computer (110) connected to an application server personal computer (120) via a network (25). The application server personal computer (120) is in turn connected via the network (25) to a database-server personal computer (130).

Please change the paragraph that begins on line 13 of page 11 to read as shown below:

The database-server personal computer (130) has, a hard drive (131) for storage of the design system database (10), operating factors database (15), process management system database (30), optionally, the simulation program database (35), and the Owner Value and RiskMap⊕ System database (45), a keyboard (132), a CRT display (133), a communications bus (134) and a read/write random access memory (135), a mouse (136), a CPU (137), and a printer (138).

Please change the paragraph that begins on line 21 of page 13 to read as shown below:

The flow diagrams in FIG. 5A and FIG. 5B detail the processing that is completed by the portion of the application software (200) that extracts, aggregates and stores the information required for system operation from: a process management system database (30), optionally, a

simulation program database (35), the Internet (40) and an Owner Value and RiskMap® System database (45) and the user (20). A brief overview of the different databases will be presented before reviewing each step of processing completed by this portion (200) of the application software.

Please change the paragraph that begins on line 1 of page 15 to read as shown below:

The Owner Value and RiskMap™ System database (45) for an enterprise contains the matrix of value, matrix of risk and statistics generated by the system described in the cross referenced application 09/994,720 dated November 28, 2001 and for a multi-enterprise organization it is the matrix of value, matrix of risk and statistics generated by the system detailed in cross-referenced application 09/994,739 dated November 28, 2001.

Please change Table 1 on page 15 to read as shown below:

<u>Table 1</u>

1. Process owner
2. Mode of operation (continuous or batch)
3. Metadata standard
4. Process resource and feature map
5. Location of process management system database and metadata ~~(optional)~~
6. Location of simulation system databases and metadata (optional)
7. Location of external database and metadata ~~(optional)~~
8. Location of Owner Value and RiskMap® System database and metadata (optional)
9. Location of Owner basic financial system and metadata (optional)
10. Location of Owner advanced financial system and metadata (optional)
11. Location of Owner operation system and metadata (optional)
12. Location of Owner asset system(s) and metadata (optional)
13. Scenario (combined normal, extreme is default)
14. Location of account structure
15. Base currency
16. Risk free cost of capital
17. Risk adjusted cost of capital
18. Management report types (text, graphic, both)
19. Default reports
20. Default missing data procedure
21. Maximum time to wait for user input
22. Maximum number of generations to process without improving fitness

Please change the paragraph that begins on line 36 of page 15 to read as shown below:

The specification of the location and metadata information for the process management system database, simulation database, external database and Owner Value and RiskMap® System database are optional because that information may have been included in the xsd and/or xml information attached to each system and data element. In which case, the software in this block would be able to locate the required data without the user (20) having to specify its metadata standard and location. After the storage of system settings data is complete, processing advances to a software block 203.

Please change the paragraph that begins on line 5 of page 16 to read as shown below:

The software in block 203 prompts the user (20) via the metadata and conversion rules window (902) to map all relevant metadata using the standard specified by the user (20) from the process management system database (30), optionally, a simulation program database (35), the Internet (40) and an Owner Value and RiskMap® System database (45) to the process resource and feature map stored in the system settings table (140). The metadata mapping specifications are saved in the metadata mapping table (141).

Please change the paragraph that begins on line 30 of page 17 to read as shown below:

After the software in block 208 initializes the bots for every mapped field within the process management system database (30) by frame, the bots extract and convert data in accordance with their preprogrammed instructions. After the extracted and converted data is stored in the process management system database table (144), processing advances to a software block 223.222.

Please delete the paragraphs and tables that begin on line 4 of page 18 as shown below:

The software in block 222 checks the bot date table (149) and deactivates any Owner Value Map® System data bots with creation dates before the current system date and retrieves

information from the system settings table (140), metadata mapping table (141), the conversion rules table (142) and the frame definition table (143). The software in block 222 then initializes data bots for retrieving the entire matrix of value and risk for each owner as well as detailed information for each cell identified the process to owner table (146) that mapped to a process feature or resource. Bots are independent components of the application that have specific tasks to perform. In the case of Owner Value Map® System data bots, their tasks are to extract and convert data detailing the matrices of value and risk for the specified owner from a specified source and store the information in a specified location. Each data bot initialized by software block 222 will store its data in the Owner Value Map® Systems table (150). Every Owner Value Map® System data bot contains the information shown in Table 3.

<center>Table 3</center>

1. Unique ID number (based on date, hour, minute, second of creation)
2. The data source location
3. Mapping information
4. Timing of extraction
5. Owner
6. Process
7. Frame
8. Segment of value, element of value, external factor or event risk
9. Conversion rules (if any)
10. Storage location (to allow for tracking of source and destination events)
11. Creation date (date, hour, minute, second)

After the software in block 222 initializes the bots they extract and convert data in accordance with their preprogrammed instructions by frame. After the extracted and converted data is stored in the Owner Value Map® Systems table (150) by frame, processing advances to a software block 223.

Please change the Table designation in lines 16 and 17 of page 18 as shown below:

simulation program data bot contains the information shown in Table 3.4.

Table 34

Please change the Table designation in lines 17 and 18 of page 19 as shown below:

table (150). Every external factor price data bot contains the information shown in Table 4.5.

Table 45

Please change the paragraph that begins on line 1 of page 21 to read as shown below:

The software in block 232 compares the data in the process management system database table (144), the simulation program table (148), the Owner Value and RiskMap® System Table (150) and the external factor forecast table (152) to determine if there any periods where required data is missing for any process. If data is missing for any process, then processing advances to a software block 233.234. Alternatively, if the required data is present for every process for every time period, then processing advances to a software block 234.302.

Please change the paragraph that begins on line 9 of page 21 to read as shown below:

The software in block 233234 prompts the user (20) via the missing process data window (907) to input the missing data displayed on the window. The new information supplied by the user (20) is stored in the appropriate table before processing advances to software block 234.

Please add the following text after line 12 of page 21 to read as shown below:

The software in block 234 checks the application database (50) to see if the Owner Value and Risk System data are current. If the data are current, then processing advances to a software block 222. If the data are not current, then processing advances to a software block 342.

The software in block 222 checks the bot date table (149) and deactivates any Owner Value and Risk System data bots with creation dates before the current system date and retrieves information from the system settings table (140), metadata mapping table (141), the conversion

rules table (142) and the frame definition table (143). The software in block 222 then initializes data bots for retrieving the entire matrix of value and risk for each owner as well as detailed information for each cell identified the process to owner table (146) that mapped to a process feature or resource. Bots are independent components of the application that have specific tasks to perform. In the case of Owner Value and Risk System data bots, their tasks are to extract and convert data detailing the matrices of value and risk for the specified owner from a specified source and store the information in a specified location. Each data bot initialized by software block 222 will store its data in the Owner Value and Risk Systems table (150). Every Owner Value and Risk System data bot contains the information shown in Table 5.

Table 5

1. Unique ID number (based on date, hour, minute, second of creation)
2. The data source location
3. Mapping information
4. Timing of extraction
5. Owner
6. Process
7. Frame
8. Segment of value, element of value, external factor or event risk
9. Conversion rules (if any)
10. Storage location (to allow for tracking of source and destination events)
11. Creation date (date, hour, minute, second)

After the software in block 222 initializes the bots they extract and convert data in accordance with their preprogrammed instructions by frame. After the extracted and converted data is stored in the Owner Value and Risk Systems table (150) by frame, processing advances to a software block 302.

Please change the paragraph that begins on line 23 of page 21 to read as shown below:

Processing in this portion of the application begins in software block 302. The software in block 302 checks the system settings table (140) in the application database (50) to determine if the current calculation is for discrete process optimization or continuous process optimization. If the process that is being optimized is a discrete process, then processing advances to a software block 322.352. Alternatively, if the process (or processes) that are being optimized is a continuous process, then processing advances to a software block 303.

Please change the paragraph that begins on line 6 of page 22 to read as shown below:

The software in block 304 retrieves the process data for the process being analyzed from the process management system database table (144) and the Owner Value and RiskMap® System table (150) before processing advances to a software block 305. The software in block 305 retrieves the process to owner mapping information for each process being analyzed from the process to owner table (146) and identifies the specific value drivers that are linked to process resource, feature and deliverables before processing advances to a software block 306. The software in block 306 retrieves the external factor prices for the process being analyzed from the external factor forecast table (152) before processing advances to a software block 307.

Please change the paragraph that begins on line 22 of page 23 to read as shown below:

The software in block 310 checks the bot date table (149) and deactivates any optimization bots with creation dates before the current system date and uses the previously retrieved information (from the system settings table (140), metadata mapping table (141), the conversion rules table (142), the frame definition table (143), the process management system database table (144), the process to owner table (146), the operating factors table (147), the simulation program table (148) - if data from there is being used - and the Owner Value and RiskMap® System table (150)). Bots are independent components of the application that have specific tasks to perform. In the case of optimization bots, their primary task is to determine the optimal mix of features and feature options for each process on a stand-alone basis by frame. The optimal mix is the mix that maximizes value and minimizes risk for the frame being analyzed. A bot for global optimization of all processes is also initiated. The optimization bots run simulations of process performance, owner risk and owner value using an unconstrained genetic algorithm that evolves to the most valuable scenario. Other optimization algorithms, including those with constraints can be used to the same effect. However, in one the preferred embodiment genetic algorithms are used. Every optimization bot activated in this block contains the information shown in Table 7.

Please change the paragraph that begins on line 23 of page 24 to read as shown below:

The software in block 311 checks the bot date table (149) and deactivates any sensitivity bots with creation dates before the current system date. The software in the block then uses the information that was previously retrieved (from the system settings table (140), metadata

mapping table (141), the conversion rules table (142), the frame definition table (143), the process management system database table (144), the process to owner table (146), the operating factors table (147), the simulation program table (148) - if data from there is being used - and the Owner Value and RiskMap® System table (150)) as required to initialize the sensitivity bots. Bots are independent components of the application that have specific tasks to perform. In the case of sensitivity bots, their primary task is to determine the sensitivity of the optimal mix to changes in element availability, external factor price, deliverable price, feature price and feature option price by process and frame. The sensitivity bots run simulations of process performance, process value and process risk using an unconstrained genetic algorithm that evolves to the most valuable scenario. Every sensitivity bot activated in this block contains the information shown in Table 8.

Please change the paragraphs that begins on line 12 of page 25 to read as shown below:
After the sensitivity bots are initialized, the bots activate in accordance with their preprogrammed instructions. After being activated, the bots determine how sensitive process value and the optimal mix of features and feature options are to changes in the process variables. The results of this analysis are saved in the sensitivity analysis table (154) in the application database (50) by process frame before processing advances to a software block 322.352.

The software in block 322352 checks the system settings table (140) in the application database (50) to determine if the current calculation is for discrete process optimization or continuous process optimization. If the process that is being optimized is a discrete process, then processing advances to a software block 323.354. Alternatively, if the process (or processes) that is being optimized is a continuous process, then processing advances to a software block 402.

The software in block 323354 checks the system settings table (140) in the application database (50) to determine if there are current calculations for all discrete process optimization items. If there are current calculations for all discrete process items, then processing advances to a software block 402. Alternatively, if there is an item (or items) that do not have current calculations, then processing advances to a software block 324.363.

The software in block 324~~363~~ retrieves data from the frame definition table (143), the process management system database table (144) and the process value table (151) as required to identify the item or items that do not have current calculations. After the software in the block identifies one or more processes without a current calculation for all frames, the software in block retrieves the complete definition of that item, the process and the frames that are associated with it from the frame definition table (143), the process management system database table (144) before processing advances to a software block 325.~~364~~.

The software in block 325~~364~~ retrieves the process data for the item being analyzed from the process management system database table (144) and the Owner Value and RiskMap® System table (150) before processing advances to a software block 326.~~365~~. The software in block 326~~365~~ retrieves the process to owner matrix mapping information for each process being analyzed from the process to owner table (146) and identifies the specific value drivers that are linked to process resource, feature and deliverables before processing advances to a software block 327.~~366~~. The software in block 327~~366~~ retrieves the external factor prices for the item and process being analyzed from the external factor forecast table (152) before processing advances to a software block 328.~~367~~.

The software in block 328~~367~~ checks the system settings table (140) to determine if simulation program data is being used in the process analysis. If simulation program data is being used, then processing advances to a software block 329.~~368~~. Alternatively, if simulation program data is not being used, then processing advances to a software block 331.~~369~~. The software in block 329~~368~~ retrieves the feature, resource and deliverable data for the process and item being analyzed from the simulation program table (148) before processing advances to software block 331.~~369~~.

The software in block 331~~369~~ checks the bot date table (149) and deactivates any feature option bots with creation dates before the current system date and retrieves information from the system settings table (140), metadata mapping table (141), the conversion rules table (142), the frame definition table (143), the process management system database table (144), the process to owner table (146), the operating factors table (147) and the simulation program table (148) if data from the latter table is being used. The software in block 331~~369~~ then initializes feature option bots by feature for the item being analyzed by process and frame. Feature option bots calculate the value the option to add a feature or remove a baseline feature by process and

frame for each item. For example, the value of the option to add piping that would facilitate a retrofit to an alternate source of water supply at a later date could be valued. The value of the real option to add or remove each feature is calculated using Black Scholes algorithms and the baseline discount rate in a manner that is well known. The real option can be valued using other algorithms including binomial, Quadranomial, neural network or dynamic programming algorithms. Feature option bots contain the information shown in Table 9.

After the feature option bots are initialized, the bots activate in accordance with their preprogrammed instructions. After being activated, the bots complete the calculation of feature option values and save the resulting values in the feature option value table (153) in the application database (50) by item before processing advances to a software block 333.~~370~~.

The software in block 333~~370~~ checks the bot date table (149) and deactivates any optimization bots with creation dates before the current system date and uses the previously retrieved information (from the system settings table (140), metadata mapping table (141), the conversion rules table (142), the frame definition table (143), the process management system database table (144), the process to owner table (146), the operating factors table (147), the simulation program table (148) - if data from there is being used - and the Owner Value and RiskMap® System table (150)). Bots are independent components of the application that have specific tasks to perform. In the case of optimization bots, their primary task is to determine the optimal mix of features and feature options for each process on a stand-alone basis by frame. The optimal mix is the mix that maximizes value and minimizes risk for the item and frame being analyzed. The optimization bots run simulations of process performance and owner value using an unconstrained genetic algorithm that evolves to the most valuable scenario. Other optimization algorithms, including those with constraints can be used to the same effect. However, in one ~~the~~ preferred embodiment genetic algorithms are used. Every optimization bot activated in this block contains the information shown in Table 10.

After the optimization bots are initialized, the bots activate in accordance with their preprogrammed instructions. After being activated, the bots determine the mix of features and feature options that optimize the process for each frame. The optimal mix is saved in the process value table (151) in the application database (50) by frame and item before processing advances to a software block 335.~~371~~.

The software in block 335371 checks the bot date table (149) and deactivates any sensitivity bots with creation dates before the current system date. The software in the block then uses the information that was previously retrieved (from the system settings table (140), metadata mapping table (141), the conversion rules table (142), the frame definition table (143), the process management system database table (144), the process to owner table (146), the operating factors table (147), the simulation program table (148) - if data from there is being used - and the Owner Value and RiskMap® System table (150)) as required to initialize the sensitivity bots. Bots are independent components of the application that have specific tasks to perform. In the case of sensitivity bots, their primary task is to determine the sensitivity of the optimal mix to changes in element availability, external factor price, deliverable price, feature price and feature option price by process and frame. The sensitivity bots run simulations of process value and process risk using an unconstrained genetic algorithm that evolves to the most valuable scenario. Every sensitivity bot activated in this block contains the information shown in Table 11.

Please insert the following after line 22, page 29.

VALUE ANALYSIS

The flow diagrams in FIG. 6C, FIG. 6D and FIG. 6E detail the processing that is completed by the portion of the application software that continually values the segments of value by enterprise. This portion of the application software also generates a matrix quantifying the impact of elements of value and external factors on the segments of value for each enterprise within the organization (see FIG. 7) by creating and activating analysis bots that:

1) Identify the factor variables, factor performance indicators and composite variables for each external factor that drive: three of the segments of value - current operation, derivatives and excess financial assets - as well as the components of current operation value (revenue, expense and changes in capital);

2) Identify the item variables, item performance indicators and composite variables for each element and sub-element of value that drive: three segments of value - current operation, derivatives and financial assets - as well as the components of current operation value (revenue, expense and changes in capital);

3) Create vectors that summarize the impact of the factor variables, factor performance indicators and composite variables for each external factor ;

4) Create vectors that summarize the performance of the item variables, item performance indicators and composite variables for each element of value and sub-element of value in driving segment value;

5) Determine the expected life of each element of value and sub-element of value;

6) Determine the current operation value, excess financial asset value and derivative value, revenue component value, expense component value and capital component value of said current operations using the information prepared in the previous stages of processing;

7) Specify and optimize causal predictive models to determine the relationship between the vectors generated in steps 3 and 4 and the three segments of value, current operation, derivatives and financial assets, as well as the components of current operation value (revenue, expense and changes in capital);

8) Determine the appropriate discount rate on the basis of relative causal element strength, value the enterprise real options and contingent liabilities and determine the contribution of each element to real option valuation;

9) Determine the best causal indicator for enterprise stock price movement, calculate market sentiment and analyze the causes of market sentiment; and

10) Combine the results of all prior stages of processing to determine the value of each element, sub-element and factor for each enterprise and the organization.

Each analysis bot generally normalizes the data being analyzed before processing begins. While the processing in the preferred embodiment includes an analysis of all five segments of value for the organization, it is to be understood that the system of the present invention can complete calculations for any combination of the five segments. For example, when a company is privately held it does not have a market price and as a result the market sentiment segment of value is not analyzed.

Processing in this portion of the application begins in software block 342. The software in block 342 aggregates, converts and stores data from the advanced financial system database (6), basic financial system database (7), operation system database (8) and one or more asset system databases (9). After data storage is complete, processing advances to a software block 343.

The software in block 343 retrieves data from the system settings table (140), the meta data mapping table (141), the asset system table (159), the element/external factor definition table

(162) and the frame definition table (143) and then assigns item variables, item performance indicators and composite variables to each element of value identified in the system settings table (140) using a three-step process. First, item variables, item performance indicators and composite variables are assigned to elements of value based on the asset management system they correspond to (for example, all item variables from a brand management system and all item performance indicators and composite variables derived from brand management system item variables are assigned to the brand element of value). Second, pre-defined composite variables are assigned to the element of value they were assigned to measure in the metadata mapping table (141). Finally, item variables, item performance indicators and composite variables identified by the text and geospatial bots are assigned to elements on the basis of their element classifications. If any item variables, item performance indicators or composite variables are un-assigned at this point they are assigned to a going concern element of value. After the assignment of variables and indicators to elements is complete, the resulting assignments are saved to the element/external factor definition table (162) by enterprise and processing advances to a block 344.

The software in block 344 retrieves data from the meta data mapping table (141), the element/external factor definition table (162) and the frame definition table (143) and then assigns factor variables, factor performance indicators and composite factors to each external factor. Factor variables, factor performance indicators and composite factors identified by the text and geospatial bots are then assigned to factors on the basis of their factor classifications. The resulting assignments are saved to element/external factor definition table (162) by enterprise and processing advances to a block 345.

The software in block 345 checks the system settings table (140) in the application database (50) to determine if any of the enterprises in the organization being analyzed have market sentiment segments. If there are market sentiment segments for any enterprise, then processing advances to a block 346. Alternatively, if there are no market prices for equity for any enterprise, then processing advances to a software block 346.

The software in block 346 checks the bot date table (149) and deactivates any market value indicator bots with creation dates before the current system date. The software in block 346 then initializes market value indicator bots in accordance with the frequency specified by the user (20) in the system settings table (140). The bot retrieves the information from the system

settings table (140), the metadata mapping table (141) and the element/external factor definition table (162) before saving the resulting information in the application database (50).

Bots are independent components of the application that have specific tasks to perform. In the case of market value indicator bots their primary task is to identify the best market value indicator (price, relative price, yield, first derivative of price change or second derivative of price change) for the time period being examined. The market value indicator bots select the best value indicator by grouping the S&P 500 using each of the five value indicators with a Kohonen neural network. The resulting clusters are then compared to the known groupings of the S&P 500. The market value indicator that produced the clusters that most closely match the know S&P 500 is selected as the market value indicator. Every market value indicator bot contains the information shown in Table 15.

Table 15

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise

When bot in block 346 have identified and stored the best market value indicator in the element/external factor definition table (162), processing advances to a block 347.

The software in block 347 checks the bot date table (149) and deactivates any temporal clustering bots with creation dates before the current system date. The software in block 347 then initializes a bot in accordance with the frequency specified by the user (20) in the system settings table (140). The bot retrieves information from the system settings table (140), the metadata mapping table (141) and the external database table (165) as required and define regimes for the enterprise market value before saving the resulting cluster information in the application database (50).

Bots are independent components of the application that have specific tasks to perform. In the case of temporal clustering bots, their primary task is to segment the market price data by

enterprise using the market value indicator selected by the bot in block 346 into distinct time regimes that share similar characteristics. The temporal clustering bot assigns a unique identification (id) number to each "regime" it identifies and stores the unique id numbers in the cluster id table (157). Every time period with data are assigned to one of the regimes. The cluster id for each regime is saved in the data record for each element variable and factor variable in the table where it resides by enterprise. If there are enterprises in the organization that don't have market sentiment calculations, then the time regimes from the primary enterprise specified by the user in the system settings table (140) are used in labeling the data for the other enterprises. After the regimes are identified, the element and factor variables for each enterprise are segmented into a number of regimes less than or equal to the maximum specified by the user (20) in the system settings table (140). The time periods are segmented for each enterprise with a market value using a competitive regression algorithm that identifies an overall, global model before splitting the data and creating new models for the data in each partition. If the error from the two models is greater than the error from the global model, then there is only one regime in the data. Alternatively, if the two models produce lower error than the global model, then a third model is created. If the error from three models is lower than from two models then a fourth model is added. The process continues until adding a new model does not improve accuracy. Other temporal clustering algorithms may be used to the same effect. Every temporal clustering bot contains the information shown in Table 16.

Table 16

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Maximum number of clusters
6. Organization
7. Enterprise

When bots in block 347 have identified and stored regime assignments for all time periods with data by enterprise, processing advances to a software block 375.

The software in block 348 checks the bot date table (149) and deactivates any variable clustering bots with creation dates before the current system date. The software in block 348 then initializes bots as required for each element of value and external factor by enterprise. The bots: activate in accordance with the frequency specified by the user (20) in the system settings table (140), retrieve the information from the system settings table (140), the metadata mapping table (141) and the element/external factor definition table (162) as required and define segments for the element variables and factor variables before saving the resulting cluster information in the application database (50).

Bots are independent components of the application that have specific tasks to perform. In the case of variable clustering bots, their primary task is to segment the element variables and factor variables into distinct clusters that share similar characteristics. The clustering bot assigns a unique id number to each "cluster" it identifies and stores the unique id numbers in the cluster id table (157). Every item variable for every element of value is assigned to one of the unique clusters. The cluster id for each variable is saved in the data record for each variable in the table where it resides. In a similar fashion, every factor variable for every external factor is assigned to a unique cluster. The cluster id for each variable is saved in the data record for the factor variable. The item variables and factor variables are segmented into a number of clusters less than or equal to the maximum specified by the user (20) in the system settings table (140). The data are segmented using the "default" clustering algorithm the user (20) specified in the system settings table (140). The system of the present invention provides the user (20) with the choice of several clustering algorithms including: an unsupervised "Kohonen" neural network, neural network, decision tree, support vector method, K-nearest neighbor, expectation maximization (EM) and the segmental K-means algorithm. For algorithms that normally require the number of clusters to be specified, the bot will iterate the number of clusters until it finds the cleanest segmentation for the data. Every variable clustering bot contains the information shown in Table 17.

Table 17

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Element of value, sub element of value or external factor
6. Clustering algorithm type
7. Organization
8. Enterprise
9. Maximum number of clusters
10. Variable 1
…to
10+n. Variable n

When bots in block 348 have identified and stored cluster assignments for the variables associated with each element of value, sub-element of value or external factor, processing advances to a software block 349.

The software in block 349 checks the bot date table (149) and deactivates any predictive model bots with creation dates before the current system date. The software in block 349 then retrieves the information from the system settings table (140), the metadata mapping table (141), the element/external factor definition table (162) and the segment definition table (176) as required to initialize predictive model bots for each component of value.

Bots are independent components of the application that have specific tasks to perform. In the case of predictive model bots, their primary task is to determine the relationship between the element and factor variables and the derivative segment of value, the excess financial asset segment of value and the current operation segment of value by enterprise. The predictive model bots also determine the relationship between the element variables and factor variables components of current operation value and sub-components of current operation value by enterprise. Predictive model bots are initialized for each component of value, sub-component of value, derivative segment and excess financial asset segment by enterprise. They are also initialized for each cluster and regime of data in accordance with the cluster and regime

assignments specified by the bots in blocks 347 and 348 by enterprise. A series of predictive model bots is initialized at this stage because it is impossible to know in advance which predictive model type will produce the "best" predictive model for the data from each commercial enterprise. The series for each model includes 12 predictive model bot types: neural network; CART; GARCH, projection pursuit regression; generalized additive model (GAM), redundant regression network; rough-set analysis, boosted Naïve Bayes Regression; MARS; linear regression; support vector method and stepwise regression. Additional predictive model types can be used to the same effect. The software in block 349 generates this series of predictive model bots for the enterprise as shown in Table 18.

Table 18

Predictive models by enterprise level
Enterprise:
Variables* relationship to enterprise cash flow (revenue – expense + capital change)
Variables* relationship to enterprise revenue component of value
Variables* relationship to enterprise expense subcomponents of value
Variables* relationship to enterprise capital change subcomponents of value
Variables* relationship to derivative segment of value
Variables* relationship to excess financial asset segment of value
Element of Value:
Sub-element of value variables relationship to element of value

*Variables = element and factor variables, item performance indicators.

Every predictive model bot contains the information shown in Table 19.

Table 19

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Global or Cluster (ID) and/or Regime (ID)
8. Segment (Derivative, Excess Financial Asset or Current Operation)
9. Element, sub-element or external factor
10. Predictive Model Type

After predictive model bots are initialized, the bots activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, the bots retrieve the required data from the appropriate table in the application database (50) and randomly partition the element or factor variables into a training set and a test set. The software in block 349 uses "bootstrapping" where the different training data sets are created by re-sampling with replacement from the original training set so data records may occur more than once. After the predictive model bots complete their training and testing, processing advances to a block 350.

The software in block 350 determines if clustering improved the accuracy of the predictive models generated by the bots in software block 349 by enterprise. The software in block 350 uses a variable selection algorithm such as stepwise regression (other types of variable selection algorithms can be used) to combine the results from the predictive model bot analyses for each type of analysis – with and without clustering - to determine the best set of variables for each type of analysis. The type of analysis having the smallest amount of error as measured by applying the mean squared error algorithm to the test data is given preference in determining the best set of variables for use in later analysis. There are four possible outcomes from this analysis as shown in Table 20.

Table 20

1. Best model has no clustering
2. Best model has temporal clustering, no variable clustering
3. Best model has variable clustering, no temporal clustering
4. Best model has temporal clustering and variable clustering

If the software in block 350 determines that clustering improves the accuracy of the predictive models for an enterprise, then processing advances to a software block 353. Alternatively, if clustering does not improve the overall accuracy of the predictive models for an enterprise, then processing advances to a software block 351.

The software in block 351 uses a variable selection algorithm such as stepwise regression (other types of variable selection algorithms can be used) to combine the results from the predictive model bot analyses for each model to determine the best set of variables for each model. The models having the smallest amount of error as measured by applying the mean squared error algorithm to the test data is given preference in determining the best set of variables. As a result of this processing, the best set of variables contain: the item variables, factor variables, item performance indicators, factor performance indications, composite variables and compoiste factors that correlate most strongly with changes in the three segments being analyzed and the three components of value. The best set of variables will hereinafter be referred to as the "value drivers". Eliminating low correlation factors from the initial configuration of the vector creation algorithms increases the efficiency of the next stage of system processing. Other error algorithms alone or in combination may be substituted for the mean squared error algorithm. After the best set of variables have been selected and stored in the element variables table (163) or factor variables table (167) for all models at all levels for each enterprise in the organization, the software in block 351 tests the independence of the value drivers at the enterprise, external factor, element and sub-element level before processing advances to a block 352.

The software in block 352 checks the bot date table (149) and deactivates any causal predictive model bots with creation dates before the current system date. The software in block 352 then retrieves the information from the system settings table (140), the metadata mapping table (141), the segment definition table (176), the element variables table (158) and the factor variables table (167) as required to initialize causal predictive model bots for each element of

value, sub-element of value and external factor in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of causal predictive model bots, their primary task is to refine the element and factor variable selection to reflect only causal variables. (Note: these variables are summed together to value an element when they are interdependent). A series of causal predictive model bots are initialized at this stage because it is impossible to know in advance which causal predictive model will produce the "best" vector for the best fit variables from each model. The series for each model includes five causal predictive model bot types: Tetrad, MML, LaGrange, Bayesian and path analysis. The software in block 352 generates this series of causal predictive model bots for each set of variables stored in the element variables table (158) and factor variables table (167) in the previous stage in processing. Every causal predictive model bot activated in this block contains the information shown in Table 21.

Table 21

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Component or subcomponent of value
6. Element, sub-element or external factor
7. Variable set
8. Causal predictive model type
9. Organization
10. Enterprise

After the causal predictive model bots are initialized by the software in block 352, the bots activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information for each model and sub-divide the variables into two sets, one for training and one for testing. After the causal predictive model bots complete their processing for each model, the software in block 352 uses a model selection algorithm to identify the model that best fits the data for each element of value, sub-

element of value and external factor being analyzed. For the system of the present invention, a cross validation algorithm is used for model selection. The software in block 352 saves the best fit causal factors in the vector table (179) by enterprise in the application database (50) and processing advances to a block 358.

The software in block 358 tests the value drivers to see if there is interaction between elements, between elements and external factors or between external factors by enterprisse. The software in this block identifies interaction by evaluating a chosen model based on stochastic-driven pairs of value-driver subsets. If the accuracy of such a model is higher that the accuracy of statistically combined models trained on attribute subsets, then the attributes from subsets are considered to be interacting and then they form an interacting set. If the software in block 358 does not detect any value driver interaction or missing variables for each enterprise, then system processing advances to a block 363. Alternatively, if missing data or value driver interactions across elements are detected by the software in block 358 for one or more enterprise, then processing advances to a software block 361.

If software in block 350 determines that clustering improves predictive model accuracy, then processing advances to block 353 as described previously. The software in block 353 uses a variable selection algorithm such as stepwise regression (other types of variable selection algorithms can be used) to combine the results from the predictive model bot analyses for each model, cluster and/or regime to determine the best set of variables for each model. The models having the smallest amount of error as measured by applying the mean squared error algorithm to the test data is given preference in determining the best set of variables. As a result of this processing, the best set of variables contains: the element variables and factor variables that correlate most strongly with changes in the components of value. The best set of variables will hereinafter be referred to as the "value drivers". Eliminating low correlation factors from the initial configuration of the vector creation algorithms increases the efficiency of the next stage of system processing. Other error algorithms alone or in combination may be substituted for the mean squared error algorithm. After the best set of variables have been selected and stored in the element variables table (158) or the factor variables table (167) for all models at all levels by enterprise, the software in block 353 tests the independence of the value drivers at the enterprise, element, sub-element and external factor level before processing advances to a block 354.

The software in block 354 checks the bot date table (149) and deactivates any causal predictive model bots with creation dates before the current system date. The software in block 354 then retrieves the information from the system settings table (140), the metadata mapping table (141), the segment definition table (176), the element variables table (158) and the factor variables table (167) as required to initialize causal predictive model bots for each element of value, sub-element of value and external factor at every level in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of causal predictive model bots, their primary task is to refine the element and factor variable selection to reflect only causal variables. (Note: these variables are grouped together to represent a single element vector when they are dependent). In some cases it may be possible to skip the correlation step before selecting causal the item variables, factor variables, item performance indicators, factor performance indicators, composite variables and composite factors. A series of causal predictive model bots are initialized at this stage because it is impossible to know in advance which causal predictive model will produce the "best" vector for the best fit variables from each model. The series for each model includes four causal predictive model bot types: Tetrad, LaGrange, Bayesian and path analysis. The software in block 354 generates this series of causal predictive model bots for each set of variables stored in the element variables table (158) in the previous stage in processing. Every causal predictive model bot activated in this block contains the information shown in Table 22.

Table 22

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Component or subcomponent of value
6. Cluster (ID) and/or Regime (ID)
7. Element, sub-element or external factor
8. Variable set
9. Organization
10. Enterprise
11. Causal predictive model type

After the causal predictive model bots are initialized by the software in block 354, the bots activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information for each model and sub-divide the variables into two sets, one for training and one for testing. The same set of training data is used by each of the different types of bots for each model. After the causal predictive model bots complete their processing for each model, the software in block 354 uses a model selection algorithm to identify the model that best fits the data for each element, sub-element or external factor being analyzed by model and/or regime by enterprise. For the system of the present invention, a cross validation algorithm is used for model selection. The software in block 354 saves the best fit causal factors in the vector table (179) by enterprise in the application database (50) and processing advances to block 358. The software in block 358 tests the value drivers to see if there are "missing" value drivers that are influencing the results as well as testing to see if there are interactions (dependencies) across elements. If the software in block 358 does not detect any missing data or value driver interactions across elements, then system processing advances to a block 363. Alternatively, if missing data or value driver interactions across elements are detected by the software in block 358, then processing advances to a software block 361.

The software in block 361 prompts the user (20) via the structure revision window (710) to adjust the specification(s) for the affected elements of value, sub-elements of value or external

factors as required to minimize or eliminate the interaction. At this point the user (20) has the option of specifying that one or more elements of value, sub elements of value and/or external factors be combined for analysis purposes (element combinations and/or factor combinations) for each enterprise where there is interaction between elements and/or factors. The user (20) also has the option of specifying that the elements or external factors that are interacting will be valued by summing the impact of their value drivers. Finally, the user (20) can chose to re-assign a value driver to a new element of value to eliminate the inter-dependency. This is the preferred solution when the inter-dependent value driver is included in the going concern element of value. Elements and external factors that will be valued by summing their value drivers will not have vectors generated. After the input from the user (20) is saved in the system settings table (140), and the element/external factor definition table (162) before system processing advances to a software block 363. The software in block 363 checks the system settings table (140) and the element/external factor definition table (162) to see if there any changes in structure. If there have been changes in the structure, then processing advances to a block 205 and the system processing described previously is repeated. Alternatively, if there are no changes in structure, then processing advances to a block 364.

The software in block 364 checks the bot date table (149) and deactivates any industry rank bots with creation dates before the current system date. The software in block 364 then retrieves the information from the system settings table (140), the metadata mapping table (141), and the vector table (179) as required to initialize industry rank bots for the enterprise and for the industry in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of industry rank bots, their primary task is to determine the relative position of each enterprise being evaluated on element variables identified in the previous processing step. (Note: these variables are grouped together when they are interdependent). The industry rank bots use ranking algorithms such as Data Envelopment Analysis (hereinafter, DEA) to determine the relative industry ranking of the enterprise being examined. The software in block 364 generates industry rank bots for each enterprise being evaluated. Every industry rank bot activated in this block contains the information shown in Table 23.

Table 23

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Ranking algorithm
6. Organization
7. Enterprise

After the industry rank bots are initialized by the software in block 364, the bots activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the item variables, item performance indicators, and composite variables from the application database (50) and sub-divides them into two sets, one for training and one for testing. After the industry rank bots develop and test their rankings, the software in block 364 saves the industry rankings in the vector table (179) by enterprise in the application database (50) and processing advances to a block 365. The industry rankings are item variables.

The software in block 365 checks the bot date table (149) and deactivates any vector generation bots with creation dates before the current system date. The software in block 365 then initializes bots for each element of value, sub-element of value and external factor for each enterprise in the organization. The bots activate in accordance with the frequency specified by the user (20) in the system settings table (140), retrieve the information from the system settings table (140), the metadata mapping table (141), the segment definition table (176) and the element variables table (158) as required to initialize vector generation bots for each element of value and sub-element of value in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of vector generation bots, their primary task is to produce formulas, (hereinafter, vectors) that summarize the relationship between the causal element variables or causal factor variables and changes in the component or sub-component of value being examined for each enterprise. The causal element variables may be grouped by element of value, sub-element of value, external factor, factor combination or element combination. As discussed previously, the

vector generation step is skipped for elements and factors where the user has specified that value driver impacts will be mathematically summed to determine the value of the element or factor. The vector generation bots use induction algorithms to generate the vectors. Other vector generation algorithms can be used to the same effect. The software in block 365 generates a vector generation bot for each set of variables stored in the element variables table (163) and factor variables table (167). Every vector generation bot contains the information shown in Table 24.

Table 24

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Element, sub-element, element combination, factor or factor combination
8. Component or sub-component of value
9. Factor 1
…to
9+n. Factor n

When bots in block 365 have identified and stored vectors for all time periods with data for all the elements, sub-elements, element combination, factor combination or external factor where vectors are being calculated in the vector table (179) by enterprise, processing advances to a software block 366.

The software in block 366 checks the bot date table (149) and deactivates any financial factor bots with creation dates before the current system date. The software in block 366 then retrieves the information from the system settings table (140), the metadata mapping table (141), the element/external factor definition table (162), the element variables table (163), the derivatives table (161), the financial forecasts table (168) and the factor variables table (167) as required to initialize causal external factor bots for the enterprise and the relevant industry in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of financial factor bots, their primary task is to identify elements of value, value drivers and external factors that are causal factors for changes in the value of: derivatives, financial assets, enterprise equity and industry equity. The causal factors for enterprise equity and industry equity are those that drive changes in the value indicator identified by the value indicator bots. A series of financial factor bots are initialized at this stage because it is impossible to know in advance which causal factors will produce the "best" model for every derivative, financial asset, enterprise or industry. The series for each model includes five causal predictive model bot types: Tetrad, LaGrange, MML, Bayesian and path analysis. Other causal predictive models can be used to the same effect. The software in block 366 generates this series of causal predictive model bots for each set of variables stored in the element variables table (163) and factor variables table (167) in the previous stage in processing by enterprise. Every financial factor bot activated in this block contains the information shown in Table 25.

Table 25

| 1. Unique ID number (based on date, hour, minute, second of creation) |
| 2. Creation date (date, hour, minute, second) |
| 3. Mapping information |
| 4. Storage location |
| 5. Element, value driver or external factor |
| 6. Organization |
| 7. Enterprise |
| 8. Type: derivatives, financial assets, enterprise equity or industry equity |
| 9. Value indicator (price, relative price, first derivative, etc.) for enterprise and industry only |
| 10. Causal predictive model type |

After the software in block 366 initializes the financial factor bots, the bots activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information and sub-divide the data into two sets, one for training and one for testing. The same set of training data is used by each of the different types of bots for each model. After the financial factor bots complete their processing for each segment of value, enterprise and industry, the software in block 366 uses a model selection

algorithm to identify the model that best fits the data for each. For the system of the present invention, a cross validation algorithm is used for model selection. The software in block 366 saves the best fit causal factors in the factor variables table (167) by enterprise and the best fit causal elements and value drivers in the element variables table (163) by enteprise and processing advances to a block 367. The software in block 367 tests to see if there are "missing" causal factors, elements or value drivers that are influencing the results by enterprise. If the software in block 367 does not detect any missing factors, elements or value drivers, then system processing advances to a block 368. Alternatively, if missing factors, elements or value drivers are detected by the software in block 367, then processing returns to software block 361 and the processing described in the preceding section is repeated.

The software in block 368 checks the bot date table (149) and deactivates any option bots with creation dates before the current system date. The software in block 368 then retrieves the information from the system settings table (140), the metadata mapping table (141), the basic financial system table (160), the external database table (165), the advanced finance system table (157) and the vector table (179) as required to initialize option bots for the enterprise.

Bots are independent components of the application that have specific tasks to perform. In the case of option bots, their primary tasks are to calculate the discount rate to be used for valuing the real options and contingent liabilities and to value the real options and contingent liabilities for the enterprise. If the user (20) has chosen to include industry options, then option bots will be initialized for industry options as well. The discount rate for enterprise real options is calculated by adding risk factors for each causal element to a base discount rate. A two step process determines the risk factor for each causal element. The first step in the process divides the maximum real option discount rate (specified by the user in system settings) by the number of causal elements. The second step in the process determines if the enterprise is highly rated on the causal elements using ranking algorithms like DEA and determines an appropriate risk factor. If the enterprise is highly ranked on the soft asset, then the discount rate is increased by a relatively small amount for that causal element. Alternatively, if the enterprise has a low ranking on a causal element, then the discount rate is increased by a relatively large amount for that causal element as shown below in Table 26.

Table 26

Maximum discount rate = 50%, Causal elements = 5		
Maximum risk factor/soft asset = 50%/5= 10%		
Industry Rank on Soft Asset		**% of Maximum**
1		0%
2		25%
3		50%
4		75%
5 or higher		100%
Causal element:	**Relative Rank**	**Risk Factor**
Brand	1	0%
Channel	3	5%
Manufacturing Process	4	7.5%
Strategic Alliances	5	10%
Vendors	2	2.5%
Subtotal		25%
Base Rate		12%
Discount Rate		37%

The discount rate for industry options is calculated using a traditional total cost of capital approach that includes the cost of risk capital in a manner that is well known. After the appropriate discount rates are determined, the value of each real option and contingent liability is calculated using the specified algorithms in a manner that is well known. The real option can be valued using a number of algorithms including Black Scholes, binomial, neural network or dynamic programming algorithms. The industry option bots use the industry rankings from prior processing block to determine an allocation percentage for industry options. The more dominant the enterprise, as indicated by the industry rank for the element indicators, the greater the allocation of industry real options. Every option bot contains the information shown in Table 27.

Table 27

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Industry or Enterprise
7. Real option type (Industry or Enterprise)
8. Real option algorithm (Black Scholes, Binomial, Quadranomial, Dynamic Program, etc.)

After the option bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information as required to complete the option valuations. When they are used, industry option bots go on to allocate a percentage of the calculated value of industry options to the enterprise on the basis of causal element strength. After the value of the real option, contingent liability or allocated industry option is calculated the resulting values are then saved in the real option value table (173) in the application database (50) by enterprise before processing advances to a block 369.

The software in block 369 checks the bot date table (149) and deactivates any cash flow bots with creation dates before the current system date. The software in the block then retrieves the information from the system settings table (140), the metadata mapping table (141), the advanced finance system table (157) and the segment definition table (176) as required to initialize cash flow bots for each enterprise in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of cash flow bots, their primary tasks are to calculate the cash flow for each enterprise for every time period where data are available and to forecast a steady state cash flow for each enterprise in the organization. Cash flow is calculated using the forecast revenue, expense, capital change and depreciation data retrieved from the advanced finance system table (157) with a well-known formula where cash flow equals period revenue minus period expense plus the period change in capital plus non-cash depreciation/amortization for the period. The steady

state cash flow for each enterprise is calculated for the enterprise using forecasting methods identical to those disclosed previously in U.S. Patent 5,615,109 to forecast revenue, expenses, capital changes and depreciation separately before calculating the cash flow. Every cash flow bot contains the information shown in Table 28.

Table 28

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise

After the cash flow bots are initialized, the bots activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated the bots, retrieve the forecast data for each enterprise from the advanced finance system table (157) and then calculate a steady state cash flow forecast by enterprise. The resulting values by period for each enterprise are then stored in the cash flow table (161) in the application database (50) before processing advances to a block 371.

The software in block 371 uses the cash flow by period data from the cash flow table (161) and the calculated requirement for working capital to calculate the value of excess financial assets for every time period by enterprise and stores the results of the calculation in the financial forecasts table (168) in the application database before processing advances to a block 372.

The software in block 372 checks the bot date table (149) and deactivates any financial value bots with creation dates before the current system date. The software in block 372 then retrieves the information from the system settings table (140), the metadata mapping table (141), the element/external factor definition table (162), the element variables table (163), the derivatives table (161) the financial forecasts table (168) and the factor variables table (167) as required to initialize financial value bots for the derivatives and excess financial assets in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of financial value bots, their primary task is to determine the relative contribution of element data and factor data identified in previous stages of processing on the value of derivatives and excess financial assets by enterprise. The system of the present invention uses 12 different types of predictive models to determine relative contribution: neural network; CART; projection pursuit regression; generalized additive model (GAM); GARCH; MMDR; redundant regression network; boosted Naïve Bayes Regression; the support vector method; MARS; linear regression; and stepwise regression. The model having the smallest amount of error as measured by applying the mean squared error algorithm to the test data is the best fit model. The "relative contribution algorithm" used for completing the analysis varies with the model that was selected as the "best-fit" as described previously. Every financial value bot activated in this block contains the information shown in Table 29.

Table 29

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Derivative or Excess Financial Asset
8. Element Data or Factor Data
9. Predictive model type

After the software in block 372 initializes the financial value bots, the bots activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information and sub-divide the data into two sets, one for training and one for testing. The same set of training data is used by each of the different types of bots for each model. After the financial bots complete their processing, the software in block 369 saves the calculated value contributions by element or external factor for derivatives in the derivatives table (161) by enterprise. The calculated value contributions by element or external factor for excess financial assets are then saved in the financial forecasts table (168) by enterprise in the application database (50) and processing advances to a block 373.

The software in block 373 checks the bot date table (149) and deactivates any element life bots with creation dates before the current system date. The software in block 373 then retrieves the information from the system settings table (140), the metadata mapping table (141) and the element/external factor definition table (162) as required to initialize element life bots for each element and sub-element of value for each enterprise in the organization being analyzed.

Bots are independent components of the application that have specific tasks to perform. In the case of element life bots, their primary task is to determine the expected life of each element and sub-element of value. There are three methods for evaluating the expected life of the elements and sub-elements of value. Elements of value that are defined by a population of members or items (such as: channel partners, customers, employees and vendors) will have their lives estimated by analyzing and forecasting the lives of the members of the population. The forecasting of member lives will be determined by the "best" fit solution from competing life estimation methods including the Iowa type survivor curves, Weibull distribution survivor curves, Gompertz-Makeham survivor curves, polynomial equations using the methodology for selecting from competing forecasts disclosed in U.S. Patent 5,615,109. Elements of value (such as some parts of Intellectual Property i.e. patents and insurance contracts) that have legally defined lives will have their lives calculated using the time period between the current date and the expiration date of the element or sub-element. Finally, elements of value and sub-element of value (such as brand names, information technology and processes) that may not have defined lives and/or that may not consist of a collection of members will have their lives estimated as a function of the enterprise Competitive Advantage Period (CAP). In the latter case, the estimate will be completed using the element vector trends and the stability of relative element strength. More specifically, lives for these element types are estimated by

1) subtracting time from the CAP for element volatility that exceeds cap volatility; and/or

2) subtracting time for relative element strength that is below the leading position and/or relative element strength that is declining;

The resulting values are stored in the element/external factor definition table (162) for each element and sub-element of value by enterprise. Every element life bot contains the information shown in Table 30.

Table 30

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Element or sub-element of value
8. Life estimation method (item analysis, date calculation or relative to CAP)

After the element life bots are initialized, they are activated in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information for each element and sub-element of value from the element/external factor definition table (162) as required to complete the estimate of element life. The resulting values are then saved in the element/external factor definition table (162) by enterprise in the application database (50) before processing advances to a block 374.

The software in block 374 checks the system settings table (140) in the application database (50) to determine if the current calculation is a new calculation or a structure change. If the calculation is not a new calculation or a structure change, then processing advances to a software block 383. Alternatively, if the calculation is new or a structure change, then processing advances to a software block 375.

The software in block 375 checks the bot date table (149) and deactivates any component capitalization bots with creation dates before the current system date. The software in block 375 then retrieves the information from the system settings table (140), the metadata mapping table (141) and the segment definition table (176) as required to initialize component capitalization bots for each enterprise in the organization.

Bots are independent components of the application that have specific tasks to perform. In the case of component capitalization bots, their task is to determine the capitalized value of the components and subcomponents of value - forecast revenue, forecast expense or forecast changes in capital for each enterprise in the organization in accordance with the formula shown in Table 31.

Table 31

$$\text{Value} = F_{f1}/(1+K) + F_{f2}/(1+K)^2 + F_{f3}/(1+K)^3 + F_{f4}/(1+K)^4 + (F_{f4} \times (1+g))/(1+K)^5) + (F_{f4} \times (1+g)^2)/(1+K)^6) \ldots + (F_{f4} \times (1+g)^N)/(1+K)^{N+4})$$

Where:

F_{fx}	= Forecast revenue, expense or capital requirements for year x after valuation date (from advanced finance system)
N	= Number of years in CAP (from prior calculation)
K	= Total average cost of capital - % per year (from prior calculation)
g	= Forecast growth rate during CAP - % per year (from advanced financial system)

After the calculation of capitalized value of every component and sub-component of value is complete, the results are stored in the segment definition table (176) by enterprise in the application database (50). Every component capitalization bot contains the information shown in Table 32.

Table 32

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Component of value (revenue, expense or capital change)
8. Sub component of value

After the component capitalization bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information for each component and sub-component of value from the advanced finance system table (157) and the segment definition table (176) as required to calculate the capitalized value of each component for each enterprise in the organization. The

resulting values are then saved in the segment definition table (176) in the application database (50) by enterprise before processing advances to a block 376.

The software in block 376 checks the bot date table (149) and deactivates any current operation bots with creation dates before the current system date. The software in block 376 then retrieves the information from the system settings table (140), the metadata mapping table (141), the element/external factor definition table (162), the segment definition table (176), the vector table (179), the financial forecasts table (168) and the factor variables table (167) as required to initialize valuation bots for each element of value, sub-element of value, combination of elements, value driver and/or external factor for the current operation.

Bots are independent components of the application that have specific tasks to perform. In the case of current operation bots, their task is to calculate the contribution of every element of value, sub-element of value, element combination, value driver, external factor and factor combination to the current operation segment of enterprise value. For calculating the current operation portion of element value, the bots use the procedure outlined in Table 5. The first step in completing the calculation in accordance with the procedure outlined in Table 5, is determining the relative contribution of each element, sub-element, combination of elements or value driver by using a series of predictive models to find the best fit relationship between:

1. The element of value vectors, element combination vectors and external factor vectors, factor combination vectors and value drivers and the enterprise components of value they correspond to; and

2. The sub-element of value vectors and the element of value they correspond to.

The system of the present invention uses 12 different types of predictive models to identify the best fit relationship: neural network; CART; projection pursuit regression; generalized additive model (GAM); GARCH; MMDR; redundant regression network; boosted Naïve Bayes Regression; the support vector method; MARS; linear regression; and stepwise regression. The model having the smallest amount of error as measured by applying the mean squared error algorithm to the test data is the best fit model. The "relative contribution algorithm" used for completing the analysis varies with the model that was selected as the "best-fit". For example, if the "best-fit" model is a neural net model, then the portion of revenue attributable to each input vector is determined by the formula shown in Table 33.

Table 33

$$\left(\sum_{k=1}^{k=m} \sum_{j=1}^{j=n} I_{jk} \times O_k \Big/ \sum_{j=1}^{j=n} I_{ik} \right) \Big/ \sum_{k=1}^{k=m} \sum_{j=1}^{j=n} I_{jk} \times O_k$$

Where

I_{jk}	= Absolute value of the input weight from input node j to hidden node k
O_k	= Absolute value of output weight from hidden node k
M	= number of hidden nodes
N	= number of input nodes

After the relative contribution of each element of value, sub-element of value, external factor, element combination, factor combination and value driver to the components of current operation value is determined, the results of this analysis are combined with the previously calculated information regarding element life and capitalized component value to complete the valuation of each: element of value, sub-element of value, external factor, element combination, factor combination and value driver using the approach shown in Table 34.

Table 34

Component Values:	Percentage	Element Life/CAP	Net Value
Revenue value = $120M	20%	80%	Value = $19.2 M
Expense value = ($80M)	10%	80%	Value = ($6.4) M
Capital value = ($5M)	5%	80%	Value = ($0.2) M
Total value = $35M			
Net value for this element:			Value = $12.6 M

The resulting values are stored in: the element/external factor definition table (162) for each element of value, sub-element of value, element combination and value driver by enterprise. For external factor and factor combination value calculations, the external factor percentage is multiplied by the capitalized component value to determine the external factor value. The resulting values for external factors are saved in the element/external factor definition table (162) by enterprise.

Every current operation bot contains the information shown in Table 35.

Table 35

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Element, sub-element, factor, element combination, factor combination or value driver
8. Component of value (revenue, expense or capital change)

After the current operation bots are initialized by the software in block 376 they activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information and complete the valuation for the segment being analyzed. As described previously, the resulting values are then saved in the element/external factor definition table (162) in the application database (50) by enterprise before processing advances to a block 377.

The software in block 377 checks the bot date table (149) and deactivates any residual bots with creation dates before the current system date. The software in block 350 then retrieves the information from the system settings table (140), the metadata mapping table (141) and the element/external factor definition table (162) as required to initialize residual bots for the each enterprise in the organization.

Bots are independent components of the application that have specific tasks to perform. In the case of residual bots, their task is to retrieve data as required from the element/external factor definition table (162) and the segment definition table (176) in order to calculate the residual going concern value for each enterprise in accordance with the formula shown in Table 36.

Table 36

Residual Going Concern Value = Total Current-Operation Value - Σ Required Financial Asset Values - Σ Elements of Value - Σ External Factors

Every residual bot contains the information shown in Table 37.

Table 37

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise

After the residual bots are initialized they activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information as required to complete the residual calculation for each enterprise. After the calculation is complete, the resulting values are then saved in the element/external factor definition table (162) by enterprise in the application database (50) before processing advances to a software block 378.

The software in block 378 determines the contribution of each element of value to the value of the real option segment of value for each enterprise. For enterprise options, the value of each element is determined by comparing the value of the enterprise options to the value that would have been calculated if the element had an average level of strength. Elements that are relatively strong, reduce the discount rate and increase the value of the option. In a similar fashion, elements that are below average in strength increase the discount rate and decrease the value of the option. The value impact can be determined by subtracting the calculated value of the option from the value of the option with the average element. The resulting values are saved in the element/external factor definition table (162) by enterprise before processing advances to block 379.

The software in block 379 checks the bot date table (149) and deactivates any sentiment calculation bots with creation dates before the current system date. The software in block 379 then retrieves the information from the system settings table (140), the metadata mapping table (141), the external database table (165), the element/external factor definition table (162), the segment definition table (176), the real option value table (173) and the derivatives table (161) as required to initialize sentiment calculation bots for the organization.

Bots are independent components of the application that have specific tasks to perform. In the case of sentiment calculation bots, their task is to retrieve data as required and then calculate the sentiment for each enterprise in accordance with the formula shown in Table 38.

Table 38

Sentiment = Market Value for Enterprise – Current Operation Value – Σ Real Option Values – Value of Excess Financial Assets - Σ Derivative Values

Enterprises that are not public corporations will, of course, not have a market value so no calculation will be completed for these enterprises. The sentiment for the organization will be calculated by subtracting the total for each of the five segments of value for all enterprises in the organization from the total market value for all enterprises in the organization. Every sentiment calculation bot contains the information shown in Table 39.

Table 39

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Type: Organization or Enterprise

After the sentiment calculation bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information from the system settings table (140), the external database table (165), the element/external factor definition table (162), the segment definition table (176), the real option value table (173), the derivatives table (161) and the financial forecasts table (168) as required to complete the sentiment calculation for each enterprise and the organization. After the calculation is complete, the resulting values are then saved in the enterprise sentiment table (164) in the application database (50) before processing advances to a block 380.

The software in block 380 checks the bot date table (149) and deactivates any sentiment analysis bots with creation dates before the current system date. The software in block 380

then retrieves the information from the system settings table (140), the metadata mapping table (141), the external database table (165), the industry ranking table (170), the element/external factor definition table (162), the segment definition table (176), the real option value table (173), the vector table (179) and the enterprise sentiment table (164) as required to initialize sentiment analysis bots for the enterprise.

Bots are independent components of the application that have specific tasks to perform. In the case of sentiment analysis bots, their primary task is to determine the composition of the calculated sentiment for each enterprise in the organization and the organization as a whole. One part of this analysis is completed by comparing the portion of overall market value that is driven by the different elements of value as determined by the bots in software block 366 and the calculated valuation impact of each element of value on the segments of value as shown below in Table 40.

Table 40

Total Enterprise Market Value = $100 Billion, 10% driven by Brand factors
Implied Brand Value = $100 Billion X 10% = $10 Billion
Brand Element Current Operation Value = $6 Billion
Increase/(Decrease) in Enterprise Real Option Values* Due to Brand = $1.5 Billion
Increase/(Decrease) in Derivative Values due to Brands = $0.0
Increase/(Decrease) in excess Financial Asset Values due to Brands = $0.25 Billion
Brand Sentiment = $10 - $6 - $1.5 - $0.0 - $0.25 = $2.25 Billion

* includes allocated industry options when used in the calculation

The sentiment analysis bots also determine the impact of external factors on sentiment. Every sentiment analysis bot contains the information shown in Table 41.

Table 41

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. External factor or element of value
6. Organization
7. Enterprise

After the sentiment analysis bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information from the system settings table (140), the metadata mapping table (141), the industry ranking table (170), the element/external factor definition table (162), the segment definition table (176), the real option value table (173), the enterprise sentiment table (164), the derivatives table (161) and the financial forecasts table (168) as required to analyze sentiment. The resulting breakdown of sentiment is then saved in the enterprise sentiment table (164) by enterprise in the application database (50). Sentiment at the organization level is calculated by adding together the sentiment calculations for all the enterprises in the organization. The results of this calculation are also saved in the enterprise sentiment table (164) in the application database (50) before processing advances to a software block 383 where the risk analysis for the organization is started.

RISK ANALYSIS

The flow diagram in FIG. 6F details the processing that is completed by the portion of the application software that analyzes and develops the matrix of risk (FIG. 7) for each enterprise in the organization. The matrix of risk includes two types of risk – the risk associated with volatility in the elements and factors driving enterprise value and the risk associated with events like hurricanes and competitor actions.

System processing in this portion of the application software (400) begins in a block 383. The software in block 383 checks the system settings table (140) in the application database (50) to determine if the current calculation is a new calculation or a structure change. If the calculation is not a new calculation or a structure change, then processing advances to a software block

392. Alternatively, if the calculation is new or a structure change, then processing advances to a software block 384.

The software in block 384 checks the bot date table (149) and deactivates any statistical bots with creation dates before the current system date. The software in block 384 then retrieves the information from the system settings table (140), the external database table (165), the element/external factor definition table (162), the element variables table (163) and the factor variables table (167) as required to initialize statistical bots for each causal value driver and external factor.

Bots are independent components of the application that have specific tasks to perform. In the case of statistical bots, their primary tasks are to calculate and store statistics such as mean, median, standard deviation, slope, average period change, maximum period change, variance and covariance for each causal value driver and external factor for all value drivers and external factors. Covariance with the market as a whole is also calculated for each value driver and external factor. Every statistical bot contains the information shown in Table 42.

Table 42

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Value driver, element variable or factor variable

When bots in block 384 have identified and stored statistics for each causal value driver and external factor in the statistics table (178) by enterprise, processing advances to a software block 385.

The software in block 385 checks the bot date table (149) and deactivates any risk reduction activity bots with creation dates before the current system date. The software in block 385 then retrieves the information from the system settings table (140), the external database table (165), the element/external factor definition table (162), the element variables table (163), the factor

variables table (167) and the statistics table (178) as required to initialize risk reduction activity bots for each causal value driver and external factor.

Bots are independent components of the application that have specific tasks to perform. In the case of risk reduction activity bots, their primary tasks are to identify actions that can be taken by the enterprise to reduce risk. For example, if one customer presents a significant risk to the enterprise, then the risk reduction bot might identify a reduction in the credit line for that customer to reduce the risk. Every risk reduction activity bot contains the information shown in Table 43.

Table 43

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Value driver or external factor

When bots in block 385 have identified and stored risk reduction activities in the risk reduction activity/product table (174) by enterprise, processing advances to a software block 386.

The software in block 386 checks the bot date table (149) and deactivates any extreme value bots with creation dates before the current system date. The software in block 386 then retrieves the information from the system settings table (140), the external database table (165), the element/external factor definition table (162), the element variables table (163) and the factor variables table (167) as required to initialize extreme value bots in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of extreme value bots, their primary task is to identify the extreme values for each causal value driver and external factor by enterprise. The extreme value bots use the Blocks method and the peak over threshold method to identify extreme values. Other extreme value algorithms can be used to the same effect. Every extreme value bot activated in this block contains the information shown in Table 44.

Table 44

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Method: blocks or peak over threshold
8. Value driver or external factor

After the extreme value bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information and determine the extreme value range for each value driver or external factor. The bot saves the extreme values for each causal value driver and external factor in the statistics table (178) by enterprise in the application database (50) and processing advances to a block 387.

The software in block 387 checks the bot date table (149) and deactivates any forecast bots with creation dates before the current system date. The software in block 386 then retrieves the information from the system settings table (140), the external database table (165), the advanced finance system table (157), the element/external factor definition table (162), the element variables table (163), the financial forecasts table (168) and the factor variables table (167) as required to initialize forecast bots in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of forecast bots, their primary task is to compare the forecasts stored for external factors and financial asset values with the information available from futures exchanges. Every forecast bot activated in this block contains the information shown in Table 45.

Table 45

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. External factor or financial asset
8. Forecast time period

After the forecast bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information and determine if any forecasts need to be changed to bring them in line with the market data on future values. The bot saves the updated forecasts in the appropriate tables in the application database (50) by enterprise and processing advances to a block 388.

The software in block 388 checks the bot date table (149) and deactivates any scenario bots with creation dates before the current system date. The software in block 388 then retrieves the information from the system settings table (140), the operation system table (171), the external database table (165), the advanced finance system table (157), the element/external factor definition table (162) and the statistics table (178) as required to initialize scenario bots in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of scenario bots, their primary task is to identify likely scenarios for the evolution of the causal value drivers and external factors by enterprise. The scenario bots use information from the advanced finance system, external databases and the forecasts completed in the prior stage to obtain forecasts for specific value drivers and factors before using the covariance information stored in the statistics table (178) to develop forecasts for the other causal value drivers and factors under normal conditions. They also use the extreme value information calculated by the previous bots and stored in the statistics table (178) to calculate extreme scenarios. Every scenario bot activated in this block contains the information shown in Table 46.

Table 46

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Type: normal or extreme
6. Organization
7. Enterprise

After the scenario bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information and develop a variety of scenarios as described previously. After the scenario bots complete their calculations, they save the resulting scenarios in the scenarios table (175) by enterprise in the application database (50) and processing advances to a block 389.

The software in block 389 checks the bot date table (149) and deactivates any simulation bots with creation dates before the current system date. The software in block 388 then retrieves the information from the system settings table (140), the operation system table (171), the advanced finance system table (157), the element/external factor definition table (162), the external database table (165), the statistics table (178), the scenarios table (175) and the generic risk table (169) as required to initialize simulation bots in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of simulation bots, their primary task is to run three different types of simulations for the enterprise. The simulation bots run simulations of organizational financial performance and valuation using: the two types of scenarios generated by the scenario bots – normal and extreme, they also run an unconstrained genetic algorithm simulation that evolves to the most negative value. In addition to examining the economic factors that were identified in the previous analysis, the bots simulate the impact of event risks like fire, earthquakes, floods and other weather-related phenomena that are largely un-correlated with the economic scenarios. Event risks are as the name implies events that may have adverse financial impacts. They generally have a range of costs associated with each occurrence. For example, every time someone

slips and falls in the factor it costs $2,367 for medical bills and lost time. The information on frequency and cost associated with these events is typically found in risk management systems. However, as discussed previously, external databases (25) may also contain information that is useful in evaluating the likelihood and potential damage associated with these risks. Event risks can also be used to project the risk associated with competitor actions, government legislation and market changes. Every simulation bot activated in this block contains the information shown in Table 47.

Table 47

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Type: normal, extreme or genetic algorithm
6. Risk factors: economic variability or event
7. Segment of value: current operation, real options, financial assets, derivatives or market sentiment
8. Organization
9. Enterprise

After the simulation bots are initialized, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). Once activated, they retrieve the required information and simulate the financial performance and value impact of the different scenarios on each segment of value by enterprise. After the simulation bots complete their calculations, the resulting risk forecasts are saved in the simulations table (177) and the xml summary table (166) by enterprise in the application database (50) and processing advances to a block 392.

The software in block 392 checks the system settings table (140) in the application database (50) to determine if the current calculation is a new calculation or a structure change. If the calculation is not a new calculation or a structure change, then processing advances to a software block 502. Alternatively, if the calculation is new or a structure change, then processing advances to a software block 393.

The software in block 393 continually runs an analysis to define the optimal risk reduction strategy for the normal and extreme scenarios for each enterprise in the organization. It starts this process by retrieving data from the system settings table (140), the operation system table (171), the external database table (165), the advanced finance system table (157), the element/external factor definition table (162), the statistics table (178), the scenarios table (175) and the risk reduction activity/product table (174) by enterprise. The software in the block determines the optimal mix of risk reduction products (derivative purchase, insurance purchase, etc.) and risk reduction activities (reducing credit limits for certain customers, shifting production from high risk to lower risk countries, etc.) for the company under each scenario given the confidence interval established by the user (20) in the system settings table (140) using a linear programming optimization algorithm. A multi criteria optimization is also run at this stage to determine the best mix for reducing risk under combined normal and extreme scenarios. Other optimization algorithms can be used at this point to achieve the same result. In any event, the resulting product and activity mix for each set of scenarios and the combined analysis is saved in the optimal mix table (172) and the xml summary table (166) in the application database (50) by enterprise and the revised simulations are saved in the simulations table (177) by enterprise before processing passes to a software block 392. The shadow prices from these optimizations are also stored in the risk reduction activity/product table (174) and the xml summary table (166) by enterprise for use in identifying new risk reduction products that the company may wish to purchase and/or new risk reduction activities the company may wish to develop. After the results of this optimization are stored in the application database (50) by enterprise, processing advances to a software block 394.

The software in block 394 checks the bot date table (149) and deactivates any impact bots with creation dates before the current system date. The software in block 393 then retrieves the information from the system settings table (140), the operation system table (171), the external database table (165), the advanced finance system table (157), the element/external factor definition table (162), the simulations table (177), the statistics table (178), the scenarios table (175) and the optimal mix table (172) as required to initialize value impact bots in accordance with the frequency specified by the user (20) in the system settings table (140).

Bots are independent components of the application that have specific tasks to perform. In the case of impact bots, their primary task is to determine the value impact of each risk reduction product and activity – those included in the optimal mix and those that are not - on the different scenarios by enterprise. Every impact bot contains the information shown in Table 48.

Table 48

1. Unique ID number (based on date, hour, minute, second of creation)
2. Creation date (date, hour, minute, second)
3. Mapping information
4. Storage location
5. Organization
6. Enterprise
7. Risk reduction product or activity

After the software in block 394 initializes the value impact bots, they activate in accordance with the frequency specified by the user (20) in the system settings table (140). After being activated, the bots retrieve information as required to revise the simulations of enterprise performance and determine the risk reduction impact of each product on each simulation. The resulting forecast of value impacts are then saved in the the risk reduction activity/product table (174) by enterprise as appropriate in the application database (50) before processing advances to a block 395.

The software in block 395 continually calculates the maximum enterprise value for each of the minimum risk strategies (normal, extreme and combined scenarios) defined in the previous section. The software in the block starts this process by retrieving data from the system settings table (140), the operation system table (171), the external database table (165), the advanced finance system table (157), the element/external factor definition table (162), the risk reduction activity/product table (174), the statistics table (178), the scenarios table (175), the financial forecasts table (168), the factor variables table (167) and the analysis definition table (158) as required to define and initialize a probabilistic simulation model for each scenario. The preferred embodiment of the probabilistic simulation model is a Markov Chain Monte Carlo model, however, other simulation models can be used with similar results. The model for each risk scenario is optimized using an optimization algorithm to identify the maximum enterprise value given the scenario risk profile. After the point of maximum value and minimum risk is identified for each scenario, the enterprise risk levels are increased and reduced in small increments and the optimization process is repeated until the efficient frontier for each scenario has been defined. The baseline efficient frontier is based on the scenario that combined normal

and extreme risk scenarios, however the results of all 3 sets of calculations (normal, extreme and combined) are saved in the report table (164) before processing advances to a block 247.

Please change the paragraph that begins on line 20 of page 30 to read as shown below:

If the user (20) has specified changes to the optimal mix, then the software in block 403 completes an analysis of the impact of the changes from all relevant frames using the optimization process described previously for blocks 310 and 333.~~370.~~ Other optimization algorithms can be used to the same effect. The software in block 403 also defines a probabilistic simulation model to analyze the proposed changes. One ~~The~~ preferred embodiment of the probabilistic simulation model is a Markov Chain Monte Carlo model. However, other simulation models can be used with similar results. The model is defined using the information retrieved from the analysis definition table (156) and then iterated as required to ensure the convergence of the frequency distribution of the output variables. After the calculation has been completed, the software in block 403 saves the resulting information in the analysis definition table (156). After displaying the results of the optional change analysis using the report selection window (909), the user (20) is prompted to specify which set of features and feature options – the optimal mix or the mix defined by the user (20) should be passed on to process management system. The mix selected for transmission to the process management system is stored in the process value table (151). After data storage is complete, the software in block 403 prompts the user (20) via a report selection data window (909) to designate reports for creation, display and/or printing. One report the user (20) has the option of selecting at this point shows the value of each feature or feature option to the process and frame being analyzed. The report also summarizes the factors that led to the addition or exclusion of each feature or feature option of the process as. When the analysis is a comparison to a prior analysis, the report will clearly show the impact of changing one or more features or feature options on the efficient frontier of the process owner as shown in FIG. 9. Other reports graphically display the sensitivity of the optimal mix to changes in the different features and external factor prices for the different frames. After the user (20) has completed the review of displayed reports and the input regarding reports to print has been saved in the reports table (145) processing advances to a software block 404.

Please change the paragraph that begins on line 12 of page 32 to read as shown below:

While the above description contains many specificities, these should not be construed as limitations on the scope of the invention, but rather as an exemplification of one preferred embodiment thereof. Accordingly, the scope of the invention should be determined not by the embodiment illustrated, but by the appended claims and their legal equivalents.